

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет радиофизики и компьютерных технологий
Кафедра интеллектуальных систем

Аннотация к дипломной работе

Использование серверной архитектуры
«Цифровой факультет» для решения сложных
вычислительных задач

Редько Александр Андреевич

Научный руководитель: старший преподаватель Штукатер Д. С.

Минск, 2024

РЕФЕРАТ

Дипломная работа: 63 страницы, 23 рисунка, 18 источников.

СЕРВЕР, ВИРТУАЛЬНАЯ МАШИНА, КОНТЕЙНЕР, PROXMOX, PFSENSE, CHROOT, TELEGRAM API, GOOGLE API, PYTHON, RAID, OAuth.

Объект исследования – серверные вычисления.

Цель работы – исследование и реализация серверной архитектуры на базе аппаратно-программной платформы «Цифровой факультет» для решения сложных вычислительных задач и внедрения результатов в процесс обучения студентов.

Методы исследования – компьютерное моделирование, эксперимент.

В данной работе подробно рассматриваются различные методы организации вычислений на сервере. Особое внимание уделяется сравнительному анализу с точки зрения гибкости, производительности и удобству использования серверных платформ виртуализации от разных производителей, таких как VMware, Hyper-V, Proxmox VE и OpenStack. На основе проведенного анализа разрабатывается оптимальная программная архитектура сервера, которая учитывает требования к производительности, безопасности, масштабируемости и простоте управления. Архитектура включает в себя выбор наиболее подходящей платформы виртуализации, настройку сетевой инфраструктуры, обеспечение отказоустойчивости.

На полученном сервере реализуется приложение для обучения нейронных сетей. Это приложение использует вычислительные мощности сервера для выполнения сложных математических операций, необходимых для тренировки нейронных сетей. В процессе разработки приложения были учтены особенности выбранной платформы виртуализации, для обеспечения максимальной эффективности и надежность работы.

В итоге, данная работа представляет собой комплексное исследование и практическое руководство по организации вычислений на сервере с использованием современных технологий виртуализации. Она демонстрирует, как правильно выбрать и настроить серверную платформу, а также как разработать и внедрить на ней приложение для обучения нейронных сетей, обеспечивая при этом высокий уровень безопасности и управляемости системы.

РЭФЕРАТ

Дыпломная праца: 63 старонкі, 23 малюнка, 18 крыніц.

СЕРВЕР, ВІРТУАЛЬНАЯ МАШЫНА, КАНТЭЙНЕР, PROXMOX, PFSENSE, CHROOT, TELEGRAM API, GOOGLE API, PYTHON, RAID, OAuth.

Аб'ект даследавання – серверныя вылічэнні.

Мэта працы – даследаванне і рэалізацыя сервернай архітэктury на базе апаратна-праграмнай платформы «Лічбавы факультэт» для вырашэння складаных вылічальных задач і ўкаранення вынікаў у працэс навучання студэнтаў.

Метады даследавання – камп'ютарнае мадэляванне, эксперымент.

У гэтай працы падрабязна разглядаюцца розныя метады арганізацыі вылічэнняў на серверы. Асаблівая ўвага надаецца параўнальнаму аналізу з пункту гледжання гібкасці, прадукцыйнасці і зручнасці выкарыстання серверных платформаў віртуалізацыі ад розных вытворцаў, такіх як VMware, Hyper-V, Proxmox VE і OpenStack. На аснове праведзенага аналізу распрацоўваецца аптымальная праграмная архітэктura сервера, якая ўлічвае патрабаванні да прадукцыўнасці, бяспекі, маштабавасці і прастаты кіравання. Архітэктura ўключае ў сябе выбар найбольш прыдатнай платформы віртуалізацыі, наладу сеткавай інфраструктуры, забеспячэнне адмоўнай устойлівасці.

На атрыманым серверы рэалізуецца прыкладанне для навучання нейронавых сетак. Гэта прыкладанне выкарыстоўвае вылічальныя магутнасці сервера для выканання складаных матэматычных аперацый, неабходных для трэніроўкі нейронавых сетак. У працэсе распрацоўкі прыкладання былі ўлічаны асаблівасці абранай платформы віртуалізацыі для забеспячэння максімальнай эфектыўнасці і надзейнасці працы.

У выніку, дадзеная праца ўяўляе сабой комплекснае даследаванне і практычнае кіраўніцтва па арганізацыі вылічэнняў на серверы з выкарыстаннем сучасных тэхналогій віртуалізацыі. Яна дэманструе, як правільна выбраць і наладзіць серверную платформу, а таксама як распрацаваць і ўкараніць на ёй прыкладанне для навучання нейронавых сетак, забяспечваючы пры гэтым высокі ўзровень бяспекі і кіравальнасці сістэмы.

ABSTRACT

Thesis: 63 pages, 23 figures, 18 sources.

SERVER, VIRTUAL MACHINE, CONTAINER, PROXMOX, PFSense, CHROOT, TELEGRAM API, GOOGLE API, PYTHON, RAID, OAUTH.

Research object – server computing.

Objective –to research and implement a server architecture based on the hardware and software platform "Digital Faculty" to solve complex computational tasks and integrate the results into the student learning process.

Research methods – computer modeling, experiment.

This work thoroughly examines various methods of organizing computations on a server. Special attention is given to comparative analysis in terms of flexibility, performance, and ease of use of virtualization server platforms from different vendors, such as VMware, Hyper-V, Proxmox VE, and OpenStack. Based on the conducted analysis, an optimal software architecture for the server is developed, considering requirements for performance, security, scalability, and ease of management. The architecture includes the selection of the most suitable virtualization platform, network infrastructure configuration, and ensuring fault tolerance.

An application for training neural networks is implemented on the resulting server. This application leverages the server's computing power to perform complex mathematical operations necessary for training neural networks. During the application development process, the features of the selected virtualization platform were taken into account to ensure maximum efficiency and reliability.

In conclusion, this work represents a comprehensive study and practical guide to organizing server computations using modern virtualization technologies. It demonstrates how to correctly choose and configure a server platform, as well as how to develop and implement an application for training neural networks on it, ensuring a high level of security and manageability of the system.